

ANÁLISIS DEL RETO

Andrés Cárdenas Layton, 202122083, a.cardenas

Esteban Benavides 202220429, e.benavidesvl

Requerimiento 0 (carga de datos)

Plantilla para documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de cómo abordaron la implementación del requerimiento

Entrada	estructura de datos del modelo, tamaño de archivo
Salidas	estructura de datos con los accidentes cargados
Implementado (Sí/No)	Si, grupal

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Abrir el archivo e iterar sobre cada accidente	$O(N)$
Verificar la fecha del accidente ya existe	$O(\log N)$
Si esta la fecha añadir al árbol el accidente, si no está crear la fecha y añadir al árbol	$O(\log N)$
TOTAL	$O(N) + O(\log N)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)/memoria kb
Small	492.47/4900.37
5pct	2171.03/20042.23
10pct	3792.38/37829.52
20pct	7602.218/73230.24
30pct	10773.78/108640.844
50pct	17367.08/179399.77
80pct	27700.19/285555.86
large	34255.66/356354.56

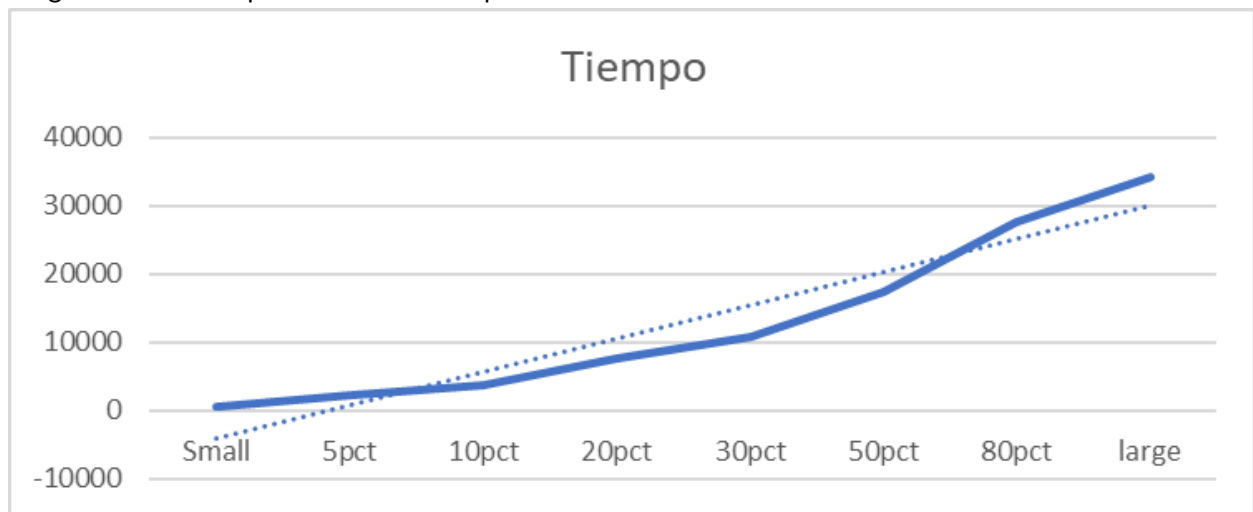
Tablas de datos

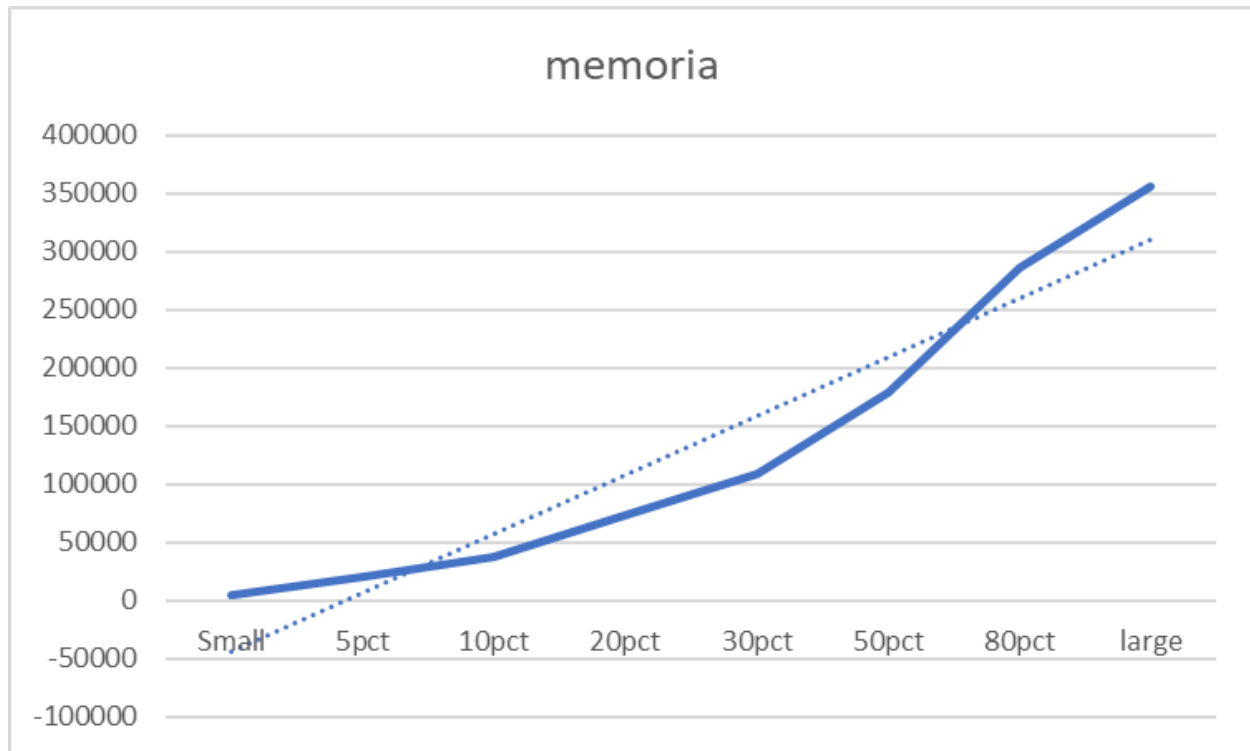
Las tablas con la recopilación de datos de las pruebas.

Entrada	Tiempo	memoria
Small	492,47	4900,37
5pct	2171,03	20042,23
10pct	3792,38	37829,52
20pct	7.602	73230,24
30pct	10773,78	108.641
50pct	17367,08	179399,77
80pct	27.700	285555,86
large	34255,66	356354,56

Gráficas

Las gráficas con la representación de las pruebas realizadas.





Análisis

Para la carga de datos se utiliza principalmente un árbol RBT el cual garantiza una complejidad de $\log N$ para la mayoría de operaciones, de la misma manera el árbol permite hacer uso de .values para futuros requerimientos, facilitando el desarrollo de estos. Sin embargo, como se utilizan loops para generar la carga de datos, esta no tiene complejidad de $\log N$ sino de $O(N) + O(\log N)$

Requerimiento 1

Plantilla para documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de cómo abordaron la implementación del requerimiento

Entrada	estructura de datos, fecha de inicio y fecha final
Salidas	Accidentes ocurridos dentro del rango de fechas
Implementado (Sí/No)	Si, grupal

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
acceder a los valores entre el rango de fechas dado, utilizando .values()	$O(\log N)$
iterar sobre cada accidente dentro del rango	$O(N^2)$
ordenar los datos por fecha	$O(N \log N)$
Poner los datos en una lista nativa para tabular	$O(N)$
TOTAL	$O(\log N) + O(N) + O(N^2) + O(N \log N)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)/memoria (kb)
Small	10.68/43.72
5pct	35.68/50.62
10pct	74.84/58.22
20pct	169.40/77.32
30pct	266.63/96.35
50pct	501.72/126.67
80pct	921.65/171.05
large	1130.30/201.03

Tablas de datos

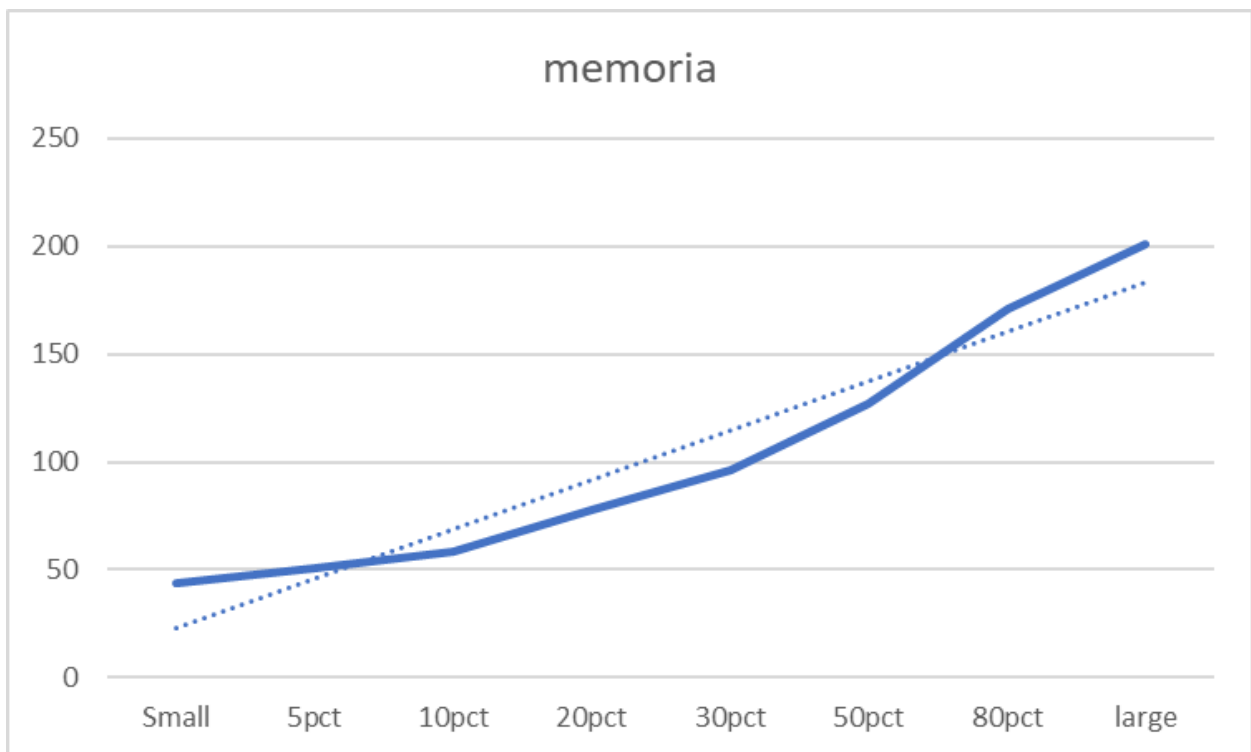
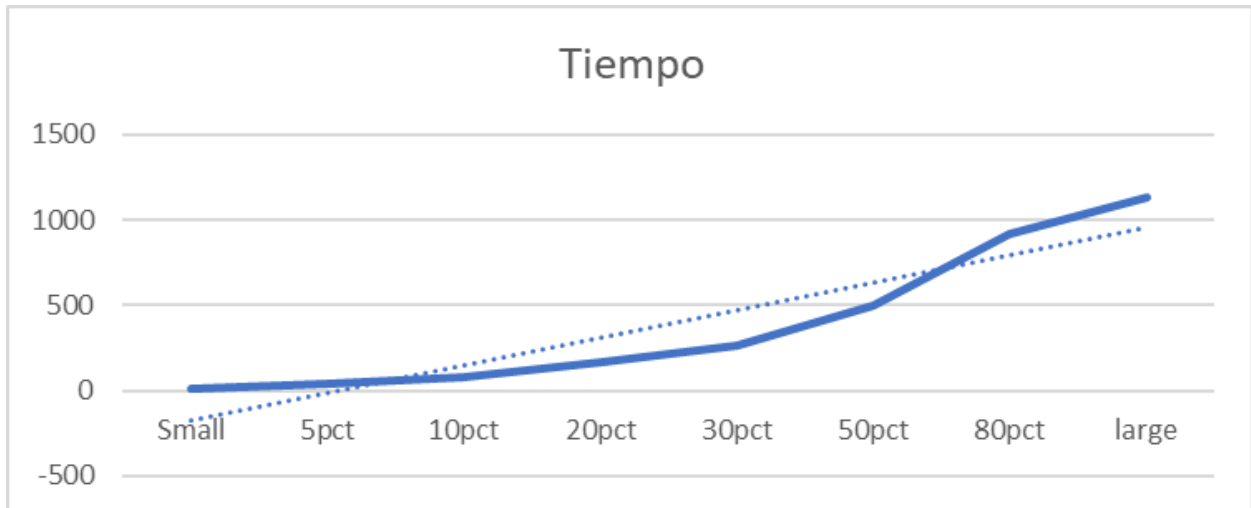
Las tablas con la recopilación de datos de las pruebas.

Entrada	Tiempo	memoria
Small	10,68	43,72
5pct	35,68	50,62
10pct	74,84	58,22
20pct	169	77,32
30pct	266,63	96
50pct	501,72	126,67
80pct	922	171,05

large 1130,3 201,03

Gráficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

Para este requerimiento se utiliza la función de .values de un RBT que permite obtener los valores dentro de un rango de llaves dado. Esta parte tiene una complejidad de $O(\log N)$, ya que se está utilizando un RBT. Sin embargo, para recorrer los accidentes dados en el rango se utiliza un ciclo dentro de un ciclo lo que genera una complejidad de $O(N^2)$, por otro lado, se utilizan ciclos individuales en el código, con el fin de recorrer diferentes elementos, por lo que también se genera una complejidad de $O(N)$. Con todas estas partes se concluye que la complejidad final del requerimiento es la suma de las distintas complejidades que se dan dentro de este.

Requerimiento 4

Plantilla para documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de cómo abordaron la implementación del requerimiento

Entrada	estructura de datos, gravedad del accidente, fecha de inicio y fecha final
Salidas	los 5 accidentes más recientes de la gravedad dada dentro del rango de fechas especificado
Implementado (Sí/No)	Si, Esteban Benavides

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
se crea un mapa nuevo que estar distribuido por gravedad y se sacan los accidentes entre el rango de fechas dado	$O(\log N)$
iterar sobre cada accidente dentro del rango añadiendo cada accidente al mapa distribuido por gravedad	$O(N^2)$
Obtener los accidentes solo de la gravedad especificada y ordenarlos por fecha	$O(N \log N)$
Poner los datos en una lista nativa para tabular	$O(N)$
TOTAL	$O(\log N) + O(N) + O(N^2) + O(N \log N)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)/memoria(kb)
Small	28.58/51.66
5pct	408.15/77.28
10pct	1277.75/101.99
20pct	5741.77/158.84
30pct	12302/170.09
50pct	37748.60/197.61
80pct	115568.67/238.68
large	177471.25/262.33

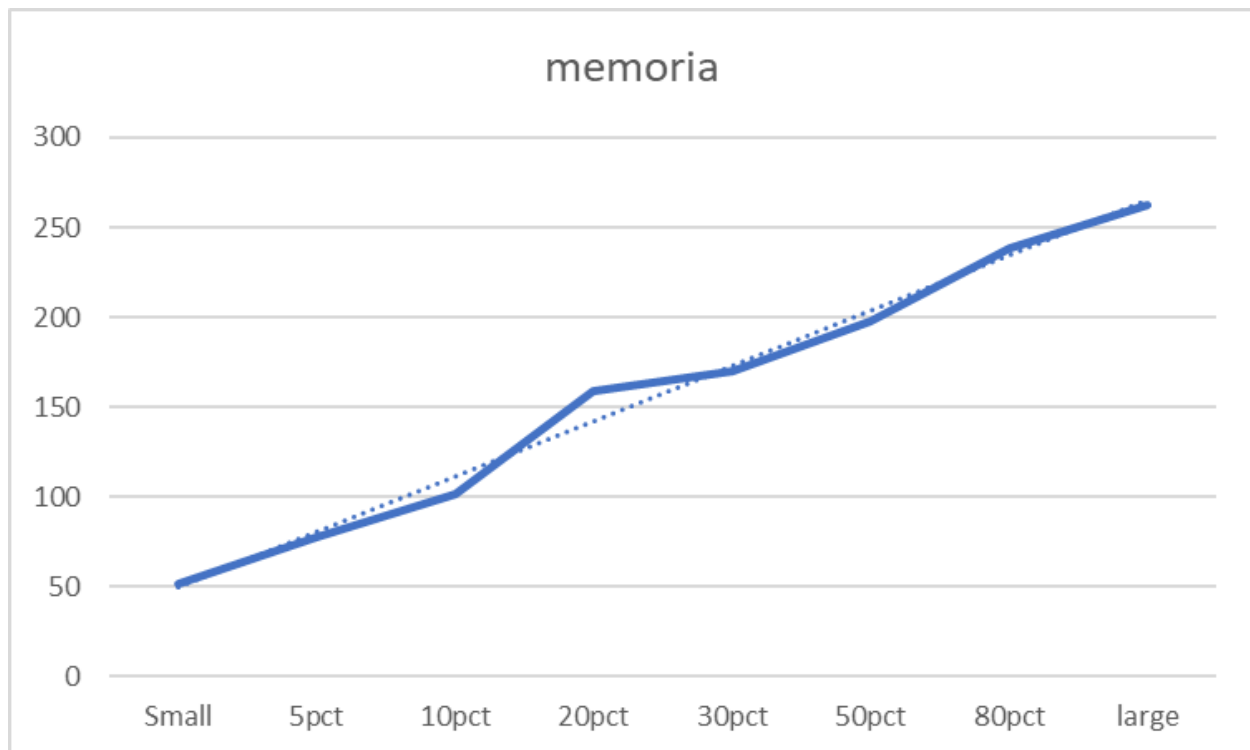
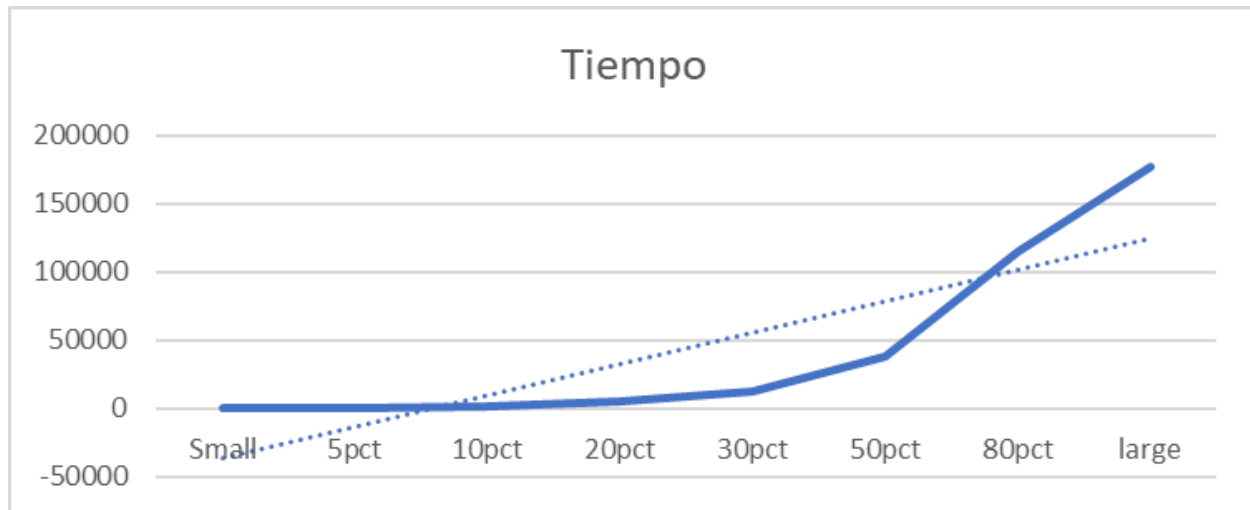
Tablas de datos

Las tablas con la recopilación de datos de las pruebas.

Entrada	Tiempo	memoria
Small	28,58	51,66
5pct	408,15	77,28
10pct	1277,75	101,99
20pct	5.742	158,84
30pct	12302	170
50pct	37748,6	197,61
80pct	115.569	238,68
large	177471,25	262,33

Gráficas

Las gráficas con la representación de las pruebas realizadas.



Análisis

Para este requerimiento, de manera similar al requerimiento 1 se utiliza la operación `.values` de un mapa ordenado, pero también se crea un nuevo mapa no ordenado en donde se guardarán los accidentes dependiendo de su gravedad, esta sección de código cuenta con una complejidad de $\log N$ gracias a que se usa un RBT. Pero de la misma manera que en el req 1 se utiliza un ciclo dentro de un ciclo para iterar sobre los accidentes, lo que genera una complejidad de $O(N^2)$. Por último al igual que en el req 1 se utiliza un ciclo independiente para iterar sobre elementos para insertarlos en una lista para tabular, de la misma manera la complejidad de este requerimiento termina siendo la suma de las complejidades de los segmentos del código

Requerimiento 5

Plantilla para documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de como abordaron la implementación del requerimiento

Entrada	estructura de datos, localidad, mes y año
Salidas	10 accidentes más recientes ocurridos en un mes y un año en una localidad de la ciudad, y el total de accidentes ocurridas en esa localidad, mes y año.
Implementado (Sí/No)	si, Andrés Cárdenas

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Se extrae el mapa de la estructura de datos, se utilizan las condicionales de todos los meses ya que varían sus días.	$O(1)$
utilizar la función datetime y la función om.values()	$O(n)$
se realizan dos recorridos de For	$O(n)^2$
se utiliza la función quicksort	$n\log(n)$
función sublist	$O(n)$
tabular los datos	$O(n)$
TOTAL	$O(1)+O(N)+O(N)^2+N\log(N)+O(N)+O(N)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)/memoria(kb)
Small	9.28/45.39
5pct	18.28/47.12
10pct	20.35/49.12
20pct	24.43/51.23
30pct	26.46/53.54
50pct	35.65/57.43
80pct	47.32/60.00
large	60.45/65.72

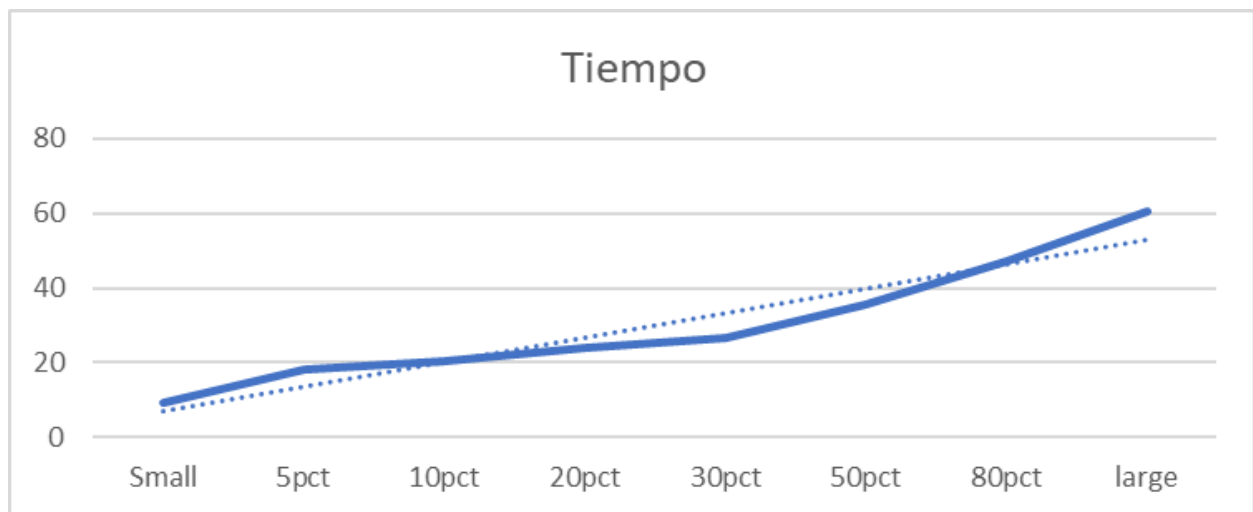
Tablas de datos

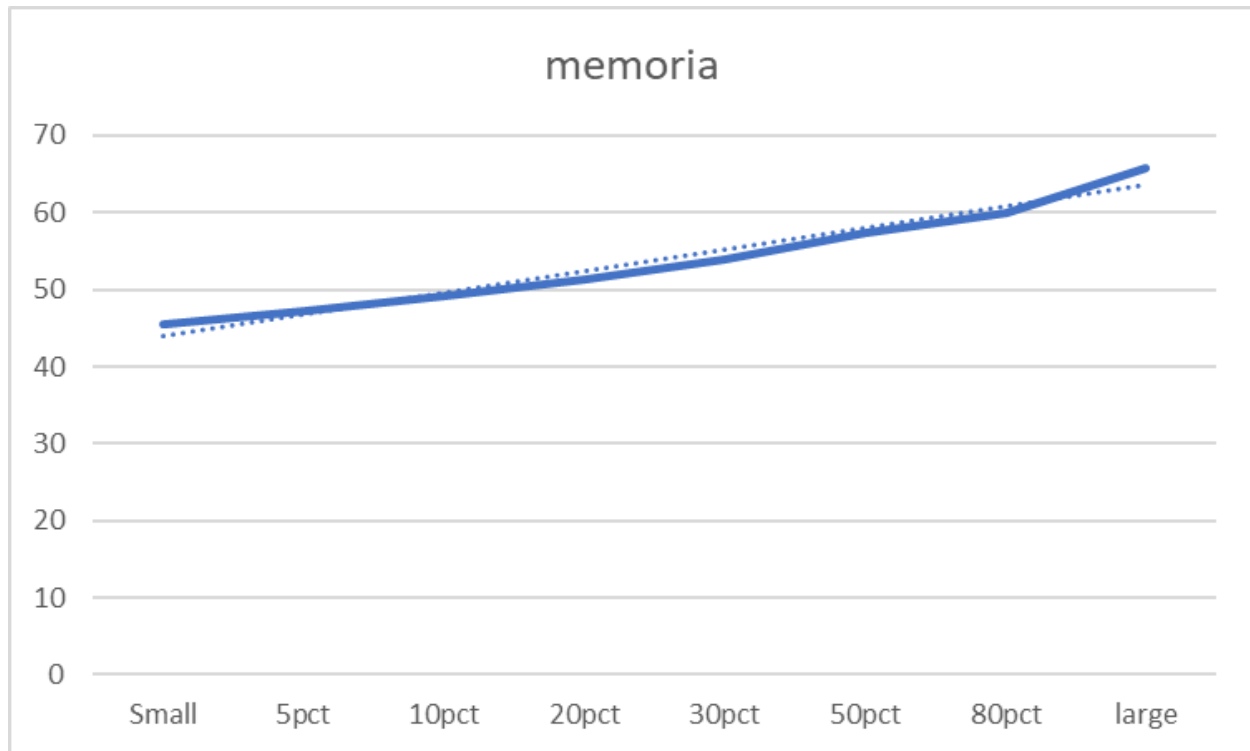
Las tablas con la recopilación de datos de las pruebas.

Entrada	Tiempo	memoria
Small	9,28	45,39
5pct	18,28	47,12
10pct	20,35	49,12
20pct	24	51,23
30pct	26,46	54
50pct	35,65	57,43
80pct	47	60
large	60,45	65,72

Gráficas

Las gráficas con la representación de las pruebas realizadas.





Análisis

Para este requerimiento primero se utilizó los parámetros de mes y año para filtrar los datos de la estructura de datos, para evitar error se hizo condicionales a partir de los días que tiene cada mes ya que luego debía transformarse con la función `datetime.strptime` en una fecha; para luego con la función de `om.values()` para así filtrar solo los datos que se encuentren en ese rango de fechas. Luego se procedió a crear una lista tipo array y recorriendo los datos filtrados (dos For) para determinar si el valor de localidad de cada dato es igual al ingresado por parámetro, este sería el último filtro para así guardar los guardar todos los datos finales en la lista array. Finalmente se organizó con QuickSort de más cercano al más lejano, para luego obtener los últimos 10 valores con ayuda de la función `sublist()` y así obtener los 10 accidentes más lejanos de esa localidad en este tiempo, y además con `Size()` se obtiene el total de accidentes ocurridos, y la complejidad $O(N)^2 + N \log(N)$ por el doble recorrido y el ordenamiento de Quick Sort.

Requerimiento 6

Plantilla para documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de cómo abordaron la implementación del requerimiento

Entrada	estructura de datos, mes, año, coordenadas (L,L), radio del área, y número de accidentes.
---------	---

Salidas	el número de accidentes ocurrido en el rango de la zona, todos seleccionados por el usuario; organizado en un tabulate
Implementado (Sí/No)	si,grupal

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
Se extrae el mapa de la estructura de datos, se utilizan las condicionales de todos los meses ya que varían sus días.	$O(1)$
función om.values()	$O(N)$
doble recorrido	$O(N)^2$
Fórmula matemática con la librería de math	$O(1)$
Función Quick Sort	$N\log(N)$
función subList()	$O(N)$
tabulate	$O(N)$
TOTAL	$O(1)+O(N)+O(N)^2+O(1)+N\log(N)+O(N)O(N)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)/memoria(kb)
Small	1.13/8.88
5pct	2.23/9.97
10pct	5.32/1.67
20pct	10.65/8.26
30pct	8.95/13.33
50pct	14.63/22.77
80pct	26.44/36.57
large	25.47/46.00

Tablas de datos

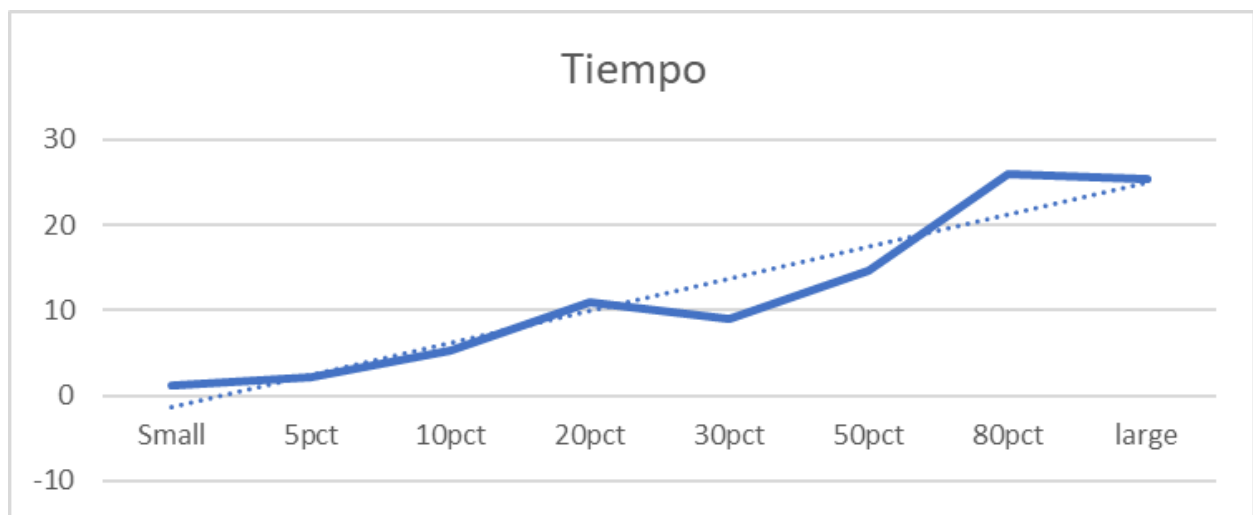
Las tablas con la recopilación de datos de las pruebas.

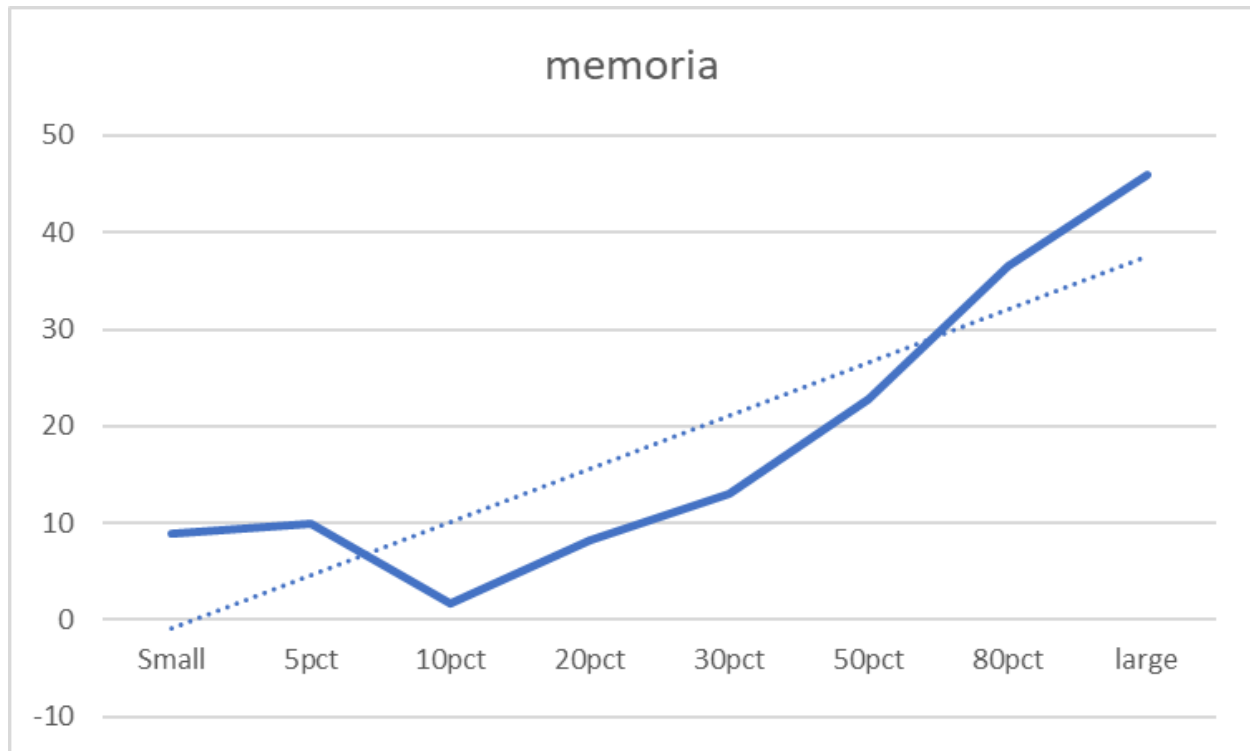
Entrada	Tiempo	memoria
Small	1,13	8,88

5pct	2,23	9,97
10pct	5,32	1,67
20pct	11	8,26
30pct	8,95	13
50pct	14,63	22,77
80pct	26	36,57
large	25,47	46

Gráficas

Las gráficas con la representación de las pruebas realizadas.





Análisis

Para este requerimiento primero se utilizó los parámetros de mes y año para filtrar los datos de la estructura de datos, para evitar error se hizo condicionales a partir de los días que tiene cada mes ya que luego debía transformarse con la función `datetime.strptime` en una fecha; para luego con la función de `om.values()` para así filtrar solo los datos que se encuentren en ese rango de fechas. Luego se procedió a crear una lista tipo array y recorriendo los datos filtrados (dos For), dentro del doble recorrido se utiliza la fórmula de de Haversine para hallar la distancia de todos los valores (se realiza una función para el cálculo con la librería `math`) de ahí se procede a crear una nueva llave en la lista y añadir las distancias a cada valor respectivo; para así finalmente hacer un filtro con condicionales almacenando en la nueva lista solo los valores que su distancia es menor al radio del área enviado por parámetro, luego se organizan los datos con Quicksort, para finalmente obtener con la función de `sublist` los elementos (de cantidad N elegidos por el usuario), y finalmente se tabula.

Requerimiento 7

Plantilla para documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de cómo abordaron la implementación del requerimiento

Entrada	estructura de datos, mes, año
---------	-------------------------------

Salidas	accidentes más temprano y masa tarde para cada día del mes y año dado, adicionalmente, un histograma de la frecuencia de los accidentes por hora para dado mes y año
Implementado (Sí/No)	Si, grupal

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
acceder a los valores dentro del mes y año dado	$O(\log N)$
Se crea una lista donde se guardan todos los accidentes de ese mes y año y se itera sobre los accidentes dentro del rango para añadirlos a la lista	$O(N^2)$
Para cada día del mes se crea una lista con sus accidentes y se ordenan por hora	$O(\log N)$
Se accede al accidente más temprano y más tarde dentro de a lista	$O(1)$
se itera sobre todos los accidentes del mes y año dado para determinar su hora y así generar los valores del histograma	$O(N)$
se itera sobre los accidentes más tempranos y más tardes de cada día para agregarlos a una lista nativa para tabular	$O(N)$
TOTAL	$O(\log N) + O(N) + O(N^2)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)/memoria(kb)
Small	5.93/6.91
5pct	38.63/8.83
10pct	125.17/8.87
20pct	216.66/8.78
30pct	349.85/8.87
50pct	718.01/9.44
80pct	1429.46/9.33
large	1987.98/8.82

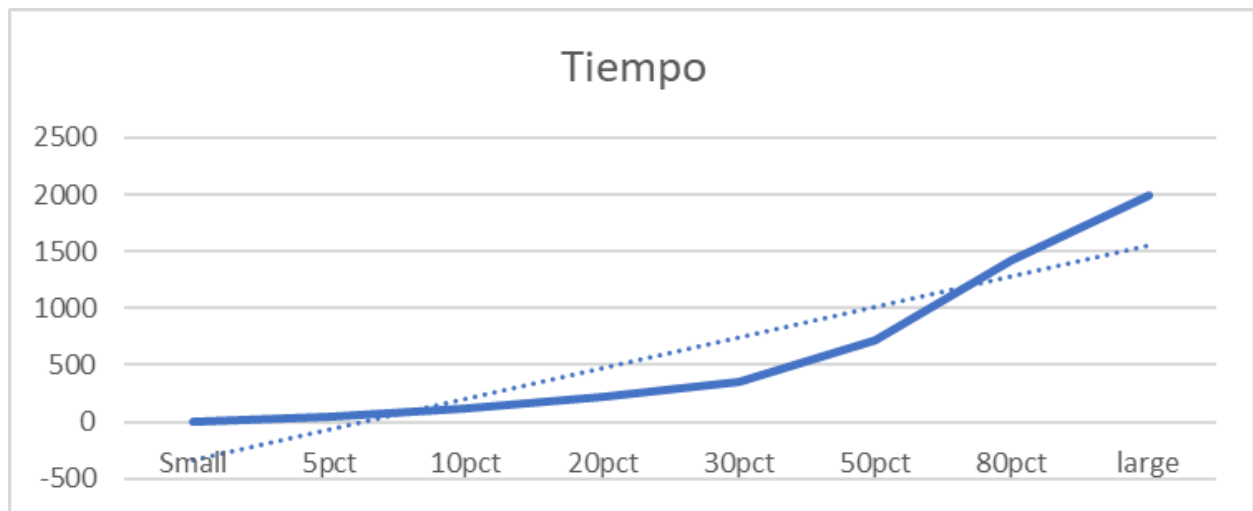
Tablas de datos

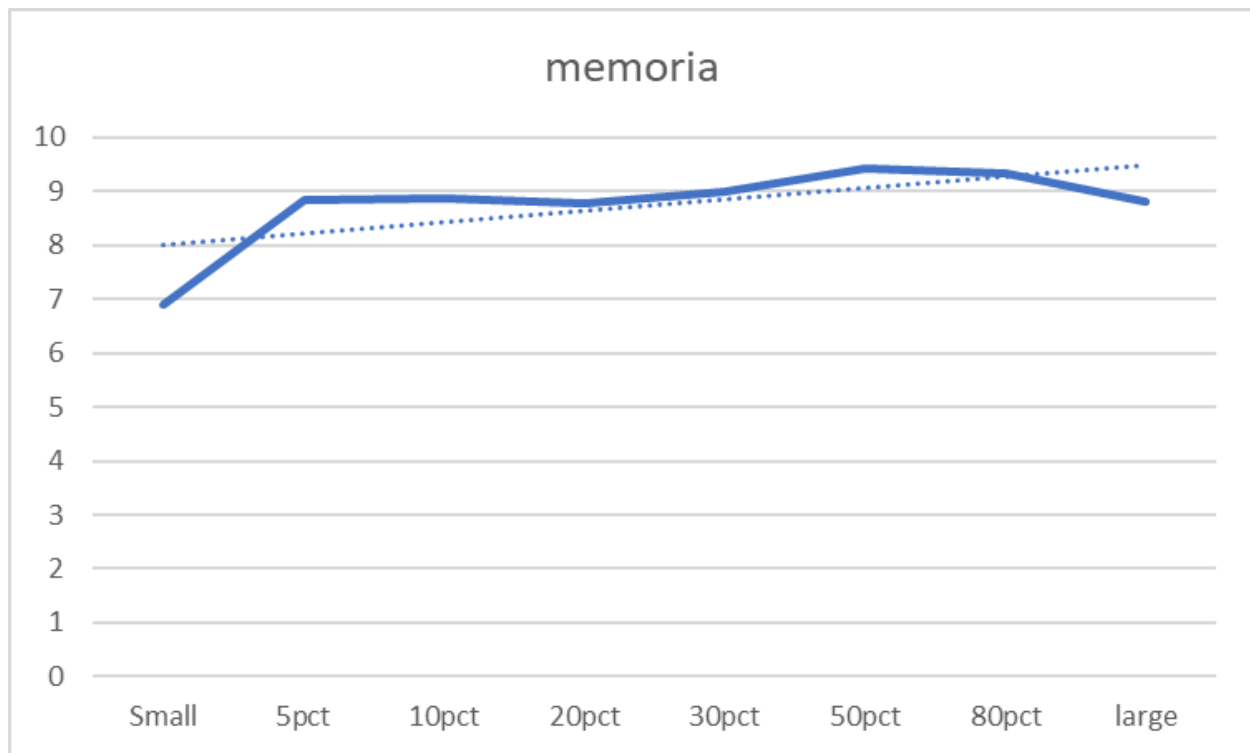
Las tablas con la recopilación de datos de las pruebas.

Entrada	Tiempo	memoria
Small	5,93	6,91
5pct	38,63	8,83
10pct	125,17	8,87
20pct	217	8,78
30pct	349,85	9
50pct	718,01	9,44
80pct	1.429	9,33
large	1987,98	8,82

Gráficas

Las gráficas con la representación de las pruebas realizadas.





Análisis

Para este requerimiento, las partes cruciales del código son prácticamente las mismas a las de los anteriores requerimientos; se obtienen los accidentes dentro de las fechas dadas, se itera sobre estos accidentes, se añaden a una lista nativa por tabular. Sin embargo, en este requerimiento, además de esto, se itera sobre cada accidente buscando su hora para poder determinar los valores a mostrar en el histograma, sin embargo, la complejidad sigue siendo la misma de los anteriores requerimientos (la suma de las complejidades de las distintas partes del código).

Requerimiento 8

Plantilla para documentar y analizar cada uno de los requerimientos.

Descripción

Breve descripción de cómo abordaron la implementación del requerimiento

Entrada	estructura de datos, fecha de inicio, fechas final y clase de accidente
Salidas	un mapa con los puntos en donde ocurrieron los accidentes de dicha clase para el rango de fechas dado.
Implementado (Sí/No)	Si, grupal

Análisis de complejidad

Análisis de complejidad de cada uno de los pasos del algoritmo

Pasos	Complejidad
se obtienen los accidentes ocurridos dentro del rango de fechas dado	$O(\log N)$
Se crea un mapa utilizando la librería "folium"	complejidad desconocida
se itera sobre cada accidente ocurrido en el rango de fechas para verificar si su clase concuerda con la especificada	$O(N^2)$
en caso de que la clase coincida, se añade un punto al mapa	complejidad desconocida
TOTAL	$O(\log N) + O(N^2)$

Pruebas Realizadas

Descripción de las pruebas de tiempos de ejecución y memoria utilizada. Incluir descripción del procedimiento, las condiciones, las herramientas y recursos utilizados (librerías, computadores donde se ejecutan las pruebas, entre otros).

Entrada	Tiempo (ms)/memoria (kb)
Small	13.46/76.99
5pct	21.88/105.80
10pct	22.65/122.03
20pct	22.19/187.94
30pct	25.99/245.94
50pct	31.75/343.70
80pct	49.32/475.69
large	50.71/566.23

Tablas de datos

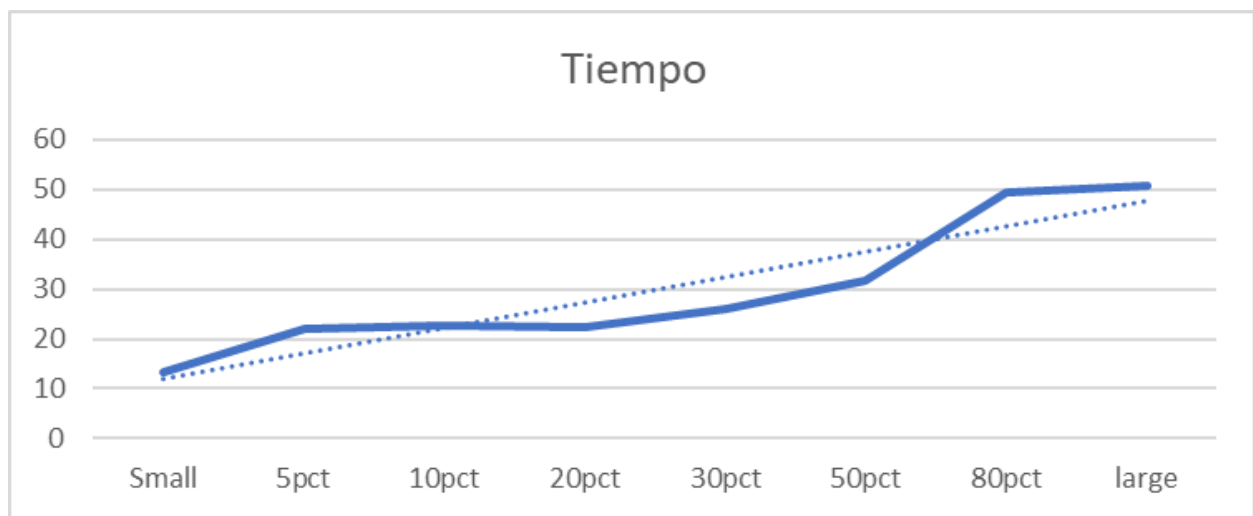
Las tablas con la recopilación de datos de las pruebas.

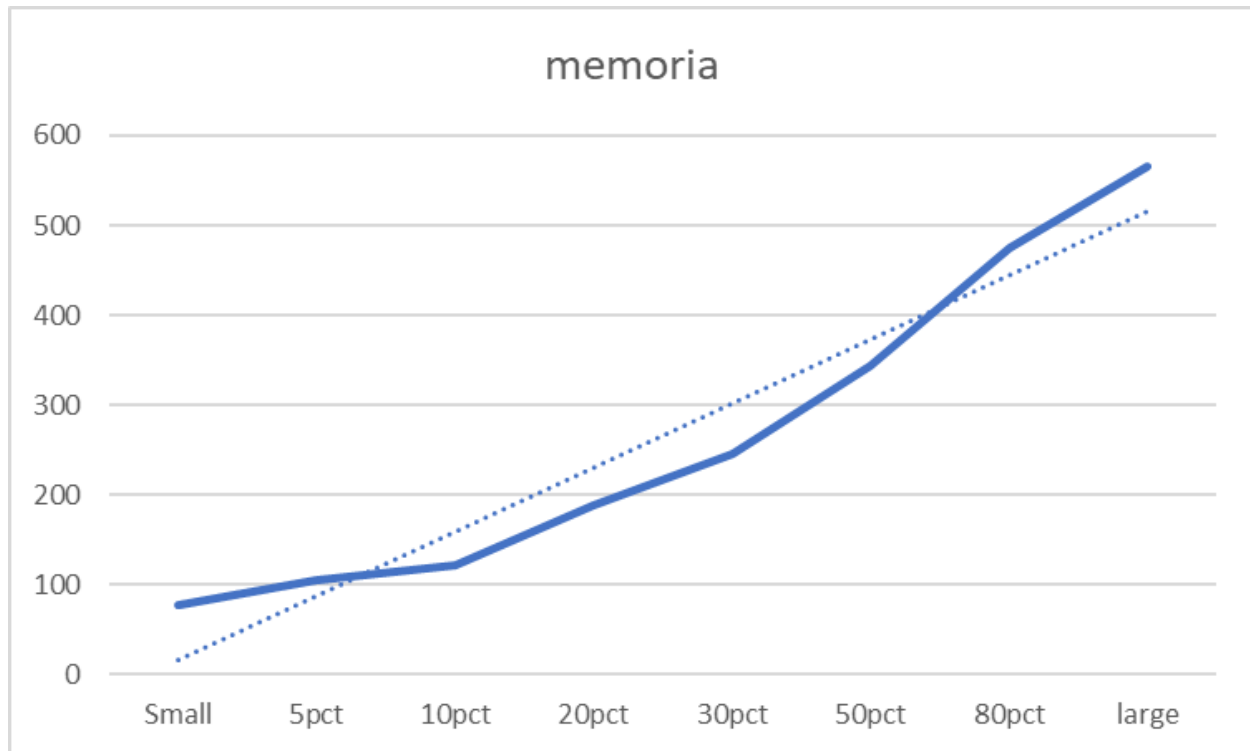
Entrada	Tiempo	memoria
Small	13,46	76,99
5pct	21,88	105,8
10pct	22,65	122,03
20pct	22	187,94

30pct	25,99	246
50pct	31,75	343,7
80pct	49	475,69
large	50,71	566,23

Gráficas

Las gráficas con la representación de las pruebas realizadas.





Análisis

Para este requerimiento es necesario comenzar de la misma manera que se hace en los otros requerimientos, encontrando los accidentes ocurridos dentro del rango de fechas, esta operación tiene una complejidad de $O(\log N)$. Al igual que en los requerimientos, se recorren estos accidentes con una complejidad de $O(N^2)$, en este caso deben ser recorridos con el fin de verificar si su clase coincide con la especificada y si este debe ser añadido al mapa. Por otro lado, en este requerimiento se utilizan operaciones de una librería externa, por lo cual se desconoce la complejidad de crear un mapa utilizando folium y de la misma manera se desconoce la complejidad de añadir un punto a dicho mapa.